プログラムコントローラの記述法* (ブール代数への追記)

村 田 昌 弥**

Description Method on the Programmable Sequence Controller* (Using Boolean Algebra and Additional New Format)

by Masaya MURATA

The Ladder Chart have been used to the presentation of programs on the Programmable Sequence Controller (P.C.) as same as on the electric sequence circuits consisted of relays. But, it is unsuitable method for the presentation of programs on P.C. because which involves new codes of micro computer never seen in sequence circuits of relays. The new description method using Boolean Algebra and the new symbols for new codes is preferred. This method gives a logical presentation and the convenience of program coding.

Key Words: Control Theory, Application of Control, Programmable Sequence Controller, Micro-Computer, Boolean Algebra, Logical Control, CAD

1. 緒 言

機械装置のシーケンス制御に、これまでのリレー回路に代わり、コンピュータを応用したプログラムコントローラ(PCと略)*^{1(1)~(6)}が上市され、電気技術者だけでなく簡単にプログラムできるようになった。

本来、シーケンス制御は論理数学(ブール代数)の応用であって、リレー回路はその一具象例にすぎない。

現在PCのプログラムは依然リレー回路に用いられてきたはし子図(ラダー図)がそのまま用いられているが、ブール代数プログラムに改めるのが本筋であらう。

しかも、PCにはコンピュータ応用の新命令語が含まれ、これはもはやラダー図ではもとより、ブール代数でさえ、端的に表現する記述法を見いだせない。

本報は、上記趣旨にもとづき、ブール代数に加えて、新命令語(コード)の代数的解釈をし、代数的意味を端的に表す記号(シンボル)と記述法を与え、さらに代数的解釈によるサブルーチンを用い、モニタを試みたものである。

2. 記号・記法・サブルーチン

記号、記法の案出に当たり、以下を考慮した。

(1) PCのコーディングはアセンブラ形式であるから、右辺に定義変数をおき、コーディング順配列にした。

(2) 関連式に関連形の表現をした。

(3) コンピュータの表現をとり入れた(積は*).

(4) 新命令語の記号・表現は、代数学で見なされた、記号、記法をとり入れた。

PCのフローは図2に示すように入力を取り込み、コード順の演算、いっせい出力変更の繰返しである。したがって、同じ回路でもコード順が変わると、結果が異なることがある。式の表現順にコードしても、不合理を生じない形式でなければならぬ。またモニタのサブルーチンも繰返しで不合理を生じてはならない。入力が1 bit 以上になる時は、サブルーチンで1 bit にする操作が必要である。以下命令語、記号、記法、モニタ用のサブルーチンについて逐時説明する。サブルーチンのメモリ割り付けは表1による。サブルーチンの言語はBASICである。

* 昭和58年7月19日 第1回設計自動化工学講演会において講演、原稿受付 昭和60年7月17日。

** 正員、信越ポリマー(株)商品研究所 (〒330 大宮市吉野町1-406)。

*1 シーケンサとも呼ぶ。

2.1 ブール代数による命令語(コード)

名 称	コード	記号	サブルーチン
読み出し (LOAD)	LD	C	
和 (OR)	OR	+	
積 (AND)	AND	*	
結 合 (BLOCK)	BLK	)	
否 定 (NOT)	NOT	$\bar{}$	
出 力 (OUT)	OUT	=	☆OU

サブルーチン ☆OU

$$M(X) = \text{SGN}(M(X)) * (1-L) :$$

$$E(X) = \text{SGN}(E(X)) * (1-L)$$

OUT には、本来サブルーチンは必要ないが、次のインタロックのため必要である。NOT には $\bar{X} = 1-X$ を代入する。NOT, BLK は他のコードと組合せる。

2.2 サブルーチンを含む命令語

2.2.1 母線切り(インタロック)コード IL

記号(開始) / (終了) ↑

メモリへの出力禁止で、自己保持力のないメモリはクリアされる。FFS, CNT, BIN, SFT はクリアされない。

サブルーチン ☆IL $L = \text{SGN}(IL)$

2.2.2 復帰(リセット)コード RST

記号 ↓ :

指定したメモリのクリア, IL の外に置く。

サブルーチン ☆RST

$$R_ (X) = 1 - \text{SGN}(R_ (X)) * (1-L)$$

2.2.3 保持(フリップフロック)コード FFS

記号 ≡

リセット以外にクリアされない自己保持

サブルーチン ☆FFS

$$F(X) = \text{SGN}(F(X) * (1-L) + FM(X)) * RF(X) :$$

$$FM(X) = F(X)$$

2.2.4 限時(タイマ)コード TM

表 1 各コードへのメモリ割り付け

Code	Address	Resistz M	Aux. Memo.
Input	A(X)		
OUT	M(X), E(X)		
IL	IL		L
FFS	F(X)		FM(X), RF(X)
TM	T(X)	T	TM(X), RT(X)
PLS	P(X)	PP(X)	PM(X)
CNT	C(X)	CP(X)	CM(X), CN(X), RC(X)
BIN	V(X)	VP(X)	VM(X), VV(X), RV(X)
SFT	SS(X) SX(X)	SP(X)	SM(X), RS(X) S1(X), S2(X), S3(X), ...

記号・記入法 入力式*時間→アドレス

入力と入力経過時間の積が指定時間(プリセット)に達すると出力される。減算形が多く、作動していない間にプリセットを読む。

サブルーチン ☆TM

$$T(X) = \text{SGN}(T(X)) * (1-L) * RT(X) :$$

$$TM(X) = T * (1-T(X)) + TM(X) * T(X)$$

$$-T(X) * \Delta T :$$

$$T(X) = (1 - \text{SGN}(TM(X)))$$

ここで ΔT はスキャン時間

2.3 パルスを応用した命令語 パルス(微分)は P.C. 特有の命令語である。スキャンを利用して、入力時・一スキャン時間だけ出力する。以下はパルスを応用している命令語である。

2.3.1 パルス(微分)コード PLS

記号 | |, 記法 | 入力式 | → アドレス

サブルーチン ☆PLS

$$PP(X) = P(X) * (1-PM(X)) * (1-L) :$$

$$PM(X) = P(X) * (1-L) : P(X) = PP(X)$$

パルスはタイマと逆に時間がくるまでの信号で、タイマ式で表現すれば、 $P(X) * \Delta T \rightarrow P(X)$ となる。

2.3.2 計数(カウンタ)コード CNT

記法 $\sum |A| > N \rightarrow C(X)$

積算値がプリセット値に達すると出力する。

サブルーチン(加算形) ☆CNT

$$CP(X) = C(X) * (1-CM(X)) * (1-L) * RC(X) :$$

$$CM(X) = C(X) * RC(X) :$$

$$CN(X) = (CN(X) + CP(X)) * RC(X) :$$

$$C(X) = \text{SGN}(CN(X) - N) + 1 :$$

$$C(X) = \text{SGN}(C(X))$$

2.3.3 交互(バイナリ)コード BIN

記号 ↔

1 bit のカウンタ、入力ごとに内容が否定され、ON-OFF を繰返すオルタネートである。

サブルーチン ☆BIN

$$VP(X) = V(X) * (1-VM(X)) * (1-L) :$$

$$VM(X) = V(X) * RV(X) :$$

$$VV(X) = \text{ABS}(VP(X) - VV(X)) :$$

$$V(X) = VV(X)$$

2.3.4 移送(シフト)コード SFT ; SET

記号および記法

シフト式 > 入力式 = アドレス → メモリ 1 → メモリ 2 (パルス) (セット) →

アドレスに入力されている内容をシフトパルスにより、メモリ 1 以下に移送する。

サブルーチン ☆SFT (Pulse)

$$SP(X)=SS(X)*(1-SM(X))*(1-L):$$

$$SM(X)=SS(X)*RS(X):$$

$$S3(X)=(S2(X)*(1-SP(X))+$$

$$S2(X)*SP(X))*RS(X):$$

$$S2(X)=(S2(X)*(1-SP(X))+$$

$$S1(X)*SP(X))*RS(X):$$

$$S1(X)=(S1(X)*(1-SP(X))+$$

$$S(X)*SP(X))*RS(X):$$

サブルーチン *SET

$$S(X)=SGN(S(X))*(1-L)*RS(X)$$

ここで $\bar{A} > A = S \rightarrow S1 \rightarrow \dots$ の動作を考えてみると、S にセットされている A が 1 であるとき、コード順が、パルスが先であれば、 $\bar{A} = 1$ (OFF) になった時、S1 へ S が転送され、1 になる。このプログラムで、 $A * \bar{A}$ (ON 後 OFF) が成立するのに、 $\bar{A} * A$ (OFF 後 ON) は成立しないから、シフトは順序交換できぬ積とされ、シーケンス制御上重要である。各コードの記法をラダー図と比較図 1 に示した。

3. パーソナルコンピュータによるモニタ

設計したプログラムが設計者の期待どおり動くか、誤動作プログラムは無いかのデバッグは極めて重要で

ある。しかし実機によるデバッグは、スキャン時間が早すぎて、プログラムの細かい演算処理状況を見たり、特に微妙な制御や、パルス応用のプログラムには不向きである。

そこで、パーソナルコンピュータで、前述の各コードに対するサブルーチンを用い、指定(アドレス)メモリに入力したうえで、サブルーチンを通す方法でスキャンごとの出力状況が見られるようにしたモニタを作った。

図 2 に、このモニタのフローを PC のフローと比較して示した。プログラム言語はベーシックを用い、変数(X)指定、入力式、サブルーチン指定の形をとった。例えば、図 3 に示すテストプログラムを例にとると、第 1 式は

$$X=0: M(0)=(A(0)+M(0))*(1-M(2))*(1-S24): GSB \quad *OU$$

と書ける。

このモニタを使えば、命令語の代数的定義さえ変わらなければ、いかなる材種の PC にも対応できる。

図 3 に、テストプログラムの例を、図 4 に、実行した例を示した。

4. 結 び

(1) PC のプログラム記述法として、ブール代数式を基本にし、新命令語を簡潔に表現する記号を案出

図 1 各コードに対する新旧記法の比較

図 2 P.C. およびモニタのフロー

LADDER CHART NEW DESCRIPTION

図3 テストプログラム

し、また、記述では、記述順が、そのままコード順になるような記述法を試みた。

この記述法では、リレーシーケンスの知識のない人でも、容易に理解し、プログラムができるのではないかと期待している。

(2) 新命令語に対する代数的解釈を明確にしたうえで、これをサブルーチン化し、入力しては、指定サブルーチンで演算する、一スキャンごとの出力表示モニタを作った。これで微妙なコントロールに対するプログラムデバッグができるのみならず、どの機種にも対応できるデバッグが可能になった。

(3) 今後の命題として、このような記述法を用いてプログラムしたうえ、直ちにデバッグ、コーディングされるCADの開発が挙げられる。

シーケンサの商品名で実機に接してから10余年になる。取扱っているうちに、プログラムをリレー回路から打ち込むことのまわりくどさ、不便利さが、このような記述法を案出するに至った。浅学非才であるが、

ニューリョフ
(00)(01)*(02)*(03)*(04)*(05)*(06)*(07)
シュツリョフ
(E0)(E1)*(E2)*(E3)*(E4)*(E5)*(E6)*(E7)

ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 1
* ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 2
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 3
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 4
* ● ** * ** * ** * ** * ** *
シュツリョフ スキャン= 5
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 6
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 7
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 8
* ● ** * ** * ** * ** * ** *
シュツリョフ スキャン= 9
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 10
* ● ** * ** * ** * ** * ** *
ニューリョフ(01)*(02)*(03)*(04)*(05)*(06)*(07)
* ** * ** * ** * ** * ** *
シュツリョフ スキャン= 11
* ● ** * ** * ** * ** * ** *
シュツリョフ スキャン= 12
* ** * ** * ** * ** * ** *

図4 テストプログラム実行例

コンピュータを応用した論理制御の考え方として、参考になればと発表した。諸賢のご批判を待つところである。

文 献

- (1) (株)日立製作所計器技術資料, No. 764-2 (昭46).
- (2) TIカタログ: 5 TIプログラムシーケンスコントローラ, (昭55).
- (3) 富士電機(株)カタログ: 産業用マイクロコントローラ, (昭55).
- (4) 立石電機(株): CZ 1000 テクニカルガイド, (昭45).
- (5) 三菱電機(株)カタログ: MELSAP, (昭59).
- (6) 光洋電子(株): KOSTAC SR 20 技術資料, (昭58).